Amendment and Response
Applicant: Renato J. Recio et al.
Serial No.: 09/980,757
Filed: May 17, 2002
Docket No.: 10002165-2
Title: PARTITIONING IN DISTRIBUTED COMPUTER SYSTEM

IN THE CLAIMS

Please cancel claims 2 and 17 without prejudice.

Please amend claims 1, 3, 7, 16, and 21 as follows:

1.   (Currently Amended) A distributed computer system comprising:

a first subnet including a first group of endnodes, each endnode in the first group of endnodes having a unique destination location identification (DLID) within the first subnet;

a second subnet including a second group of endnodes, each endnode in the second group of endnodes having a unique DLID within the second subnet;

a communication fabric physically coupled to the first group of endnodes and the second group of endnodes; and

a partitioning mechanism associating a first partition key to every DLID in the first subnet for enabling communication between the first group of endnodes over the communication fabric and associating a second partition key to every DLID in the second subnet for enabling communication between the second group of endnodes over the communication fabric, wherein the endnodes each include a partition key table having a partition key and DLIDs pertinent to the partition key.

2.   (Cancelled)

3.   (Currently Amended) The distributed computer system of claim 1̲2̶, wherein the endnodes each include non-volatile memory storing the partition key table.

4.   (Previously Presented) The distributed computer system of claim 1, wherein each subnet includes a partition manager providing definition and setup of partitions on the subnet.

5.   (Previously Presented) The distributed computer system of claim 4, wherein at least one subnet includes multiple partition managers.

2

6.   (Previously Presented) The distributed computer system of claim 4, wherein at least one partition manager spans a plurality of subnets, and each of the plurality of subnets includes a partition manager agent being controlled by the partition manager to manage partitions on the subnet.

7.   (Currently Amended) The distributed computer system of claim 1, ~~wherein the endnodes each include a partition key table having a partition key and DLIDs pertinent to the partition key,~~ wherein each subnet includes a partition manager providing definition and setup of partitions on the subnet, and wherein the partition manager controls the content of the partition key table.

8.   (Previously Presented) The distributed computer system of claim 7, wherein a new partition key value is accepted by an endnode to be placed in the partition key table only when a correct management key is passed with the new partition key value, wherein the management key is employed only for management messages.

9.   (Previously Presented) The distributed computer system of claim 8, wherein the partition manager employs the management key when the partition manager creates partitions by writing partition keys or removing an endnode from a partition by invalidating a partition key.

10.   (Previously Presented) The distributed computer system of claim 1, wherein the partitioning mechanism inserts a partition key associated to a source endnode in a frame sent from the source endnode, stores a partition key at a destination endnode, and compares the partition key in the frame to the stored partition key at the destination endnode.

11.   (Previously Presented) The distributed computer system of claim 10, wherein if the partition key in the frame matches the stored partition key at the destination endnode, the frame is accepted and processed normally.

3

12.   (Previously Presented) The distributed computer system of claim 10, wherein if the partition key in the frame does not match the stored partition key at the destination endnode, the frame is rejected.

13.   (Previously Presented) The distributed computer system of claim 10, wherein if the partition key in the frame does not match the stored partition key at the destination endnode, an error is returned to the source endnode.

14.   (Previously Presented) The distributed computer system of claim 10, wherein the partition key is stored in a frame header in a frame context area as a frame key value.

15.   (Previously Presented) The distributed computer system of claim 14, wherein the frame key value is obtained by performing a logical operation on the partition key and another type of key value.

16.   (Currently Amended) A method of partitioning endnodes in a distributed computer system, the method comprising:
        defining a first subnet including a first group of endnodes, each endnode in the first group of endnodes having a unique destination location identification (DLID) within the first subnet;
        defining a second subnet including a second group of endnodes, each endnode in the second group of endnodes having a unique DLID within the second subnet;
        physically coupling, with a communication fabric, the first group of endnodes and the second group of endnodes;
        associating a first partition key to every DLID in the first subnet for enabling communication between the first group of endnodes over the communication fabric; and
        associating a second partition key to every DLID in the second subnet for enabling communication between the second group of endnodes over the communication fabric; and
        establishing, at each endnode, a partition key table having a partition key and DLIDs pertinent to the partition key.

4

Amendment and Response
Applicant: Renato J. Recio et al.
Serial No.: 09/980,757
Filed: May 17, 2002
Docket No.: 10002165-2
Title: PARTITIONING IN DISTRIBUTED COMPUTER SYSTEM

17.   (Cancelled)

18.   (Previously Presented) The method of claim 16 further comprising:
        defining and establishing partitions on each subnet with a partition manager.

19.   (Previously Presented) The method of claim 18 further comprising:
        defining and establishing partitions on at least one subnet with multiple partition

managers.

20.   (Previously Presented) The method of claim 16 further comprising:
        managing partitions on a plurality of subnets via one partition manager which controls

a partition manager agent on each of the plurality of subnets.

21.   (Currently Amended) The method of claim 16 further comprising:
        establishing a partition key table having a partition key and DLIDs pertinent to the
partition key;
        defining and establishing partitions on each subnet with a partition manager; and
        controlling the content of the partition key table with the partition manager.

22.   (Previously Presented) The method of claim 21 further comprising:
        accepting a new partition key value at an endnode to be placed in the partition key
table only when a correct management key is passed with the new partition key value,
wherein the management key is employed only for management messages.

23.   (Previously Presented) The method of claim 22 further comprising:
        creating partitions by writing partition keys with the partition manager employing the
management key; and
        removing an endnode from a partition by invalidating a partition key with the
partition manager employing the management key.

24.   (Previously Presented) The method of claim 16 further comprising:

5

inserting a partition key associated to a source endnode in a frame sent from the source endnode;

storing a partition key at a destination endnode; and

comparing the partition key in the frame to the stored partition key at the destination endnode.

25. (Previously Presented) The method of claim 24 further comprising:

accepting the frame and processing the frame normally if the partition key in the frame matches the stored partition key at the destination endnode.

26. (Previously Presented) The method of claim 24 further comprising:

rejecting the frame if the partition key in the frame does not match the stored partition key at the destination endnode.

27. (Previously Presented) The method of claim 24 further comprising:

returning an error to the source endnode if the partition key in the frame does not match the stored partition key at the destination endnode.

28. (Previously Presented) The method of claim 24 further comprising:

storing the partition key in a frame header in a frame context area as a frame key value.

29. (Previously Presented) The method of claim 28 further comprising:

obtaining the frame key value by performing a logical operation on the partition key and another type of key value.

6